Risk Assessment and Mitigation

In every project there are potential threats that need to be managed or will massively impact production, potentially stopping it altogether. Risk management is an essential part of projects that attempts to manage these threats and stop them becoming a problem. It works in four stages, identification, analysis, mitigation and monitoring. The following document is our presentation of the likelihood, impact and mitigation of the risks to our project that we gathered from conducting these four stages of risk management.

In the identification stage we used the basic concept of the Taxonomy Based Questionnaire [1] but shortened it considerably as our project is small scale and non-critical. Tailoring it to fit 3 main categories (Management, Requirements and Production) we systematically discovered possible threats to the project. We then went back through the discovered risks and eliminated any with a very low probability or very minor consequences. We were then left with a concise list of risks by category to perform analysis on.

The next step was analysing the risks and assigning each one a likelihood and severity based on the team's judgement. Both of these were split into four classifications Low, Medium, High and Critical and colour coded. Measures were defined as is shown on the right. We added the two scores together to come up with a risk management score, the higher the score the bigger the problem, and did some further elimination by removing every risk with a score of 2.

Classification	Likelihood	Severity
Critical	$n > 15$	$n > 5$
High	$10 < n \leq 15$	$3 < n \leq 5$
Medium	$5 < n \leq 10$	$1 < n \leq 10$
Low	$n \geq 5$	$n \leq 1$

Where n = number of times the team judged it would happen in project without proper management
d = number of days work it would take to rectify the situation/solve the problem

The team then planned mitigation for each risk utilising both avoidance strategies, that reduce the likelihood, and Minimisation strategies, that reduce the severity. For example within the mitigation for the risk of "Faulty External Libraries" our avoidance strategy was "Testing the external libraries as thoroughly as the developed work" meaning bugs would be found before we use them and therefore would not carry through into our product, and the minimisation strategy was "developing using a modular design means the team can simply replace any libraries which do have a fault easily" meaning if we do use a faulty external library we don't have to start the whole product from scratch because it is dependent on it. We then gave each risk a post mitigation risk management score (calculated from predicted severity plus predicted likelihood after our mitigation strategies have been conducted) and assessed this. If the post mitigation score was too high (5 or higher) we had to remove our mitigation strategies and instead implement a contingency plan, which is a proven method that will help mitigate the risk.

Description and owner	Result	Area	Severity	Likelihood	Score	Mitigation	Post Mit Score
Members leaving Luke	Massive increase of pressure on each team member, whatever member was working on possibly lost, schedule and plan may need re-working. All apply as long as member is absent.	Management	Critical	Low	5	Multiple people assigned to any critical task so that if a member leaves there is still one member on it and up to date to catch others up if needed. If the team member is lost permanently then re-scheduling, re-planning and revising management must take place by organising a meeting ASAP. Reduces severity slightly.	4
Miscommunication between team Sam	Duplication of work, some work not being done at all, pressure put on team members relationships.	Management	High	High	6	By assigning multiple people to each task it means there will always be someone to check with at short notice/pick up on communication errors, reducing likelihood. The very regular meetings we've scheduled along with constant contact (via Messenger) means that any time it does happen it will be much less severe.	2
Productivity problems Sam	Low quality/late work being produced if any work produced at all.	Management	High	Low	4	Using Messenger keeps the team in constant contact. Reminders are set regularly to apply a bit of pressure and give a sense of urgency, whilst praising good work and making sure each member is happy and has a place to voice any issues. This will improve morale and increase productivity.	3
Failures in version control Ollie	Large errors can become very expensive in terms of time. Accidental branching.	Management	High	Medium	5	Clear version control filing system used on Google Drive - the team's regular team meetings will cover what version members are currently using to ensure everyone is up to date.	3
Errors in time estimation Luke	Schedule may need re-working, extra pressure may be put on to meet larger deadlines.	Management	Medium	Medium	4	Thorough planning and examination of the work of previous groups should reduce likelihood, with the use of Messenger and regular meetings extra members can be assigned to tasks that are taking longer than estimated	3
Failure of file hosting Ollie	Have to start over from the latest point.	Management / Production	Critical	Low	5	Offline copies of saves will be taken so in the event of a failure the latest save will only be at most a couple of days work ago.	3
Inadequate knowledge/skill level Martin	Extra time needed to gain skills if the team member cannot currently do their task.	Management / Production	High	Medium	6	By covering the skill set of each individual on the team and discussing assignment of new tasks in the team's regular meetings team members will be doing tasks that they are good at, reducing the likelihood of having an inadequate knowledge/skill level and reducing the time taken to rectify this if so.	3
Shift in deadlines Luke	Schedule may need re-working, extra pressure may be put on to meet new deadlines.	Management	Medium	Low	4	Designing the team's schedule to have critical tasks done as early as possible and leaving days before deadlines will mean that small shifts in deadlines will have no effect and large shifts will have a reduced severity.	2
Gaps in the planning of the project Luke	Some essential tasks may be discovered at an inconvenient point (e.g. just before hand in at deadline/when next task depends on it and it hasn't been done therefore delaying critical path).	Management	High	Medium	6	Team meetings and communication via messenger will ensure if a critical task is missing it will be discovered early, reducing severity. By following the Scrum framework of the agile method and frequently iterating over our plans we will massively reduce likelihood by discovering gaps in our Gantt charts long before we start work on that section	3
Gaps in management of project Sam	Work will be slower, of lower quality and may have a negative effect on team mood.	Management	High	Medium	6	Because of the frequent meetings and review structure of the scrum framework, any gaps in management will be found and rectified early and easily.	3

Change of requirements during/post development Luke	Work that fit old requirements but not new requirements is rendered useless.	Requirements	Critical	Low	5	There's nothing that can be done to stop this happening as it is out of the team's control. Frequent meetings with customer to ensure any changes are caught early will reduce severity slightly. Using a modular design will ensure that the team can use as many previously developed modules as possible and only change what is necessary will also help reduce severity.	4
Misunderstanding of customer requirements Luke	Work that fit the team's understanding of the customer requirement, but not the customers intention may be rendered useless.	Requirements	Critical	Medium	6	Having a meeting with the customer and looking over every user requirement in depth and briefly looking over system requirements means any misunderstanding is unlikely and will not be very severe. Frequently meeting with the user and showing the current progress will ensure if the team starts developing something that is straying from customer's requirements it will be caught and rectified early.	2
Clash of requirements David	Proper work cannot be produced or tested to requirements and therefore is useless.	Requirements	Critical	Low	5	Analysing the requirements and negotiating with stakeholders wherever a clash is discovered means requirements can be changed at an early stage so there are no clashes. If any changes are made to requirements further analysis (and possible negotiations) will take place.	2
Change of intended user Luke	User may have new requirements.	Requirements	High	Low	4	Meeting with the customer ASAP, discussing new user's requirements and re-covering current requirements will ensure that only essential changes will be made to requirements and they will be made as early as possible so no unnecessary work will be produced.	3
Unnecessary requirements (gold plating) Ben	May slow down production and introduce unnecessary clashes.	Requirements	Medium	Low	3	Keep requirements as only what is absolutely necessary, any gold plating can be done when core requirements have been completed to the customer's satisfaction. Extra-requirements to ensure changes are in line with customers ideas can be drafted up in a later stage meeting if necessary.	2
Ambiguity of requirements Martin	Team members understanding of the same requirement may vary and therefore they may produce work that doesn't fit together or meet the customer's requirement.	Requirements	Critical	Medium	6	By writing a fit criteria for each requirement it ensures that they are not ambiguous and can be easily measured and tested.	3
Change in user's expectations Luke	May lead to major change in requirements.	Requirements	High	Low	4	Keeping in constant contact with the customer means that changes in expectations are caught earlier and will reduce the severity of a change. Before the customer builds up an idea of a product very different to the original product specified the team can meet with the customer and negotiate requirements.	2
Underestimation of project scale Luke	More time needed for certain tasks, tools currently being used may no longer be appropriate.	Requirements	High	Low	4	Simplifying project as much as possible and using a modular design ensures best scalability.	2
Development hardware failure (power cut) Martin	Work currently being produced will be lost/changes since last save lost.	Production	Low	High	4	Saving regularly will reduce severity but there is nothing that the team can do to reduce the likelihood of an external hardware fault.	4
Unreliable final product Ollie	Customer may reject product.	Production	Critical	Medium	6	Using thorough alpha, beta and acceptance testing the team can find any bugs to ensure the final product is reliable.	3

Gaps in testing Ollie	May produce unreliable product.	Production	High	High	6	Planning, documenting and analysing tests in a spiral model ensures any gaps are found and covered before final product is implemented.	3
Faulty external libraries David	May produce unreliable product.	Production	High	Medium	5	Testing the external libraries as thoroughly as the developed work reduces the likelihood of this hazard occurring while developing using a modular design means the team can simply replace any libraries which do have a fault easily, massively reducing the severity.	3
Not matching product to requirements Ben	Customer may reject product.	Production	Critical	Low	5	Using the iterative process of the agile method and having regular meetings with customer, along with carefully designing tests that check the product meets the fit criteria keeps both the likelihood of a requirement not being met and the severity low.	2
Incompatibility between development tools and required devices David	May produce product that is incompatible with required design.	Production	Critical	Low	5	Frequently testing the product on the intended device (showing progression to customer) makes sure any tools used are compatible or caught early, and using a modular design means a section developed with incompatibility can be removed and replaced easily.	4
Basic human errors Ben	May produce larger errors further down production.	Production	Medium	Critical	6	Assigning multiple people to the same job catches errors early and means that if a member is struggling with a topic they have someone to ask before making a mistake, reducing likelihood. Using an iterative agile method during development and a modular design makes it easier to rectify errors and reduces severity.	4

Some risks have completely different severities and likelihoods depending on how long they will be in effect. For example, it is quite likely that a team member will not be available for one day, but it is unlikely to have much of an impact on production time, however, a team member leaving the team for good would have a very severe effect but is much less likely. We decided to plan for the most severe possibility after discovering Murphy's law and agreeing that if it is going to go wrong we need to plan for the risk with the largest possible impact.

The system we have for assigning scores is colour coding them is as follows:

Likelihood\Severity	Low	Medium	High	Critical
Low	2	3	4	5
Medium	3	4	5	6
High	4	5	6	7
Critical	5	6	7	8

Risks that do not require any further management/pose no threat to the project.
Risks that still pose a threat but if treated carefully should not be a problem.
Risks that pose a threat that will definitely impede production without further management.
Risks that pose a large threat, definitely impeding production for multiple days without proper management.
Risks that could potentially prevent the project from finishing, needs urgent and careful management.

References

1) CARR, M. J., KONDA, S. L., MONARCH, I., ULRICH, F. C. AND WALKER, C. F.
Taxonomy-Based Risk Identification
In-text: [1]
Your Bibliography: [1]M. Carr, S. Konda, I. Monarch, F. Ulrich and C. Walker, *Taxonomy-Based Risk Identification*. Ft. Belvoir: Defense Technical Information Center, 1993.
https://resources.sei.cmu.edu/asset_files/TechnicalReport/1993_005_001_16166.pdf